## *Mobile Application Development (CPRG-303-E)*

## Project Phase 2: App Architecture

## **Prepared by**

***Coffee Mates***

*Team Name*

Members:

## *Claur, Alessandra Nicole*

## *Escartin, Mitzi Vera*

## *Fontelo, Lulubelle*

## *Siwa, Gabriel Ira*

*Teruel, Bryan Benedict*

**Prepared for**

Mohammed Al-Haifi

Date: March 05, 2025

## 

## 

## 

## **School for Advanced Digital Technology**

## **1. Introduction**

This document details the architectural Decisions for Studdy, a mobile study planner designed to help students manage their tasks, set reminders, and track progress efficiently. This architecture plan is based on the team's skill set and project requirements, ensuring a structured development process.

## **2. Architectural Framework Overview**

## The Studdy mobile application follows a three-tier architecture, ensuring modularity, scalability, and ease of maintenance:

## **2.1. Architecture Diagram**

A diagram of a study planner

AI-generated content may be incorrect.

## **2.2 Presentation Layer (Frontend)**

## Built using React Native for cross-platform support.

## Utilizes Bootstrap for UI consistency.

## Implements React Navigation for screen transitions.

## UI components include Task List, Calendar View, Reminders, and Progress Tracker.

## **2.3 Business Logic Layer**

## Handles task management, scheduling reminders, and progress tracking.

## Ensures seamless navigation between screens using stack-based routing.

## **2.4 Data Layer (Storage & Backend)**

## Local Storage: react-native-mmkv (https://github.com/mrousavy/react-native-mmkv/tree/main/package)

## Data Flow: UI interactions → Business Logic Processing → Data Storage & Retrieval.

## **3. Architecture Decision: Records**

### **3.1 Development Framework**

**Status:** Accepted

Context: Our mobile app is a \*\*Study Planner App\*\* that allows students to create tasks, set reminders, and receive notifications. We need a framework that supports cross-platform mobile app development with Android as the target device.

**Decision:**

We will use \*\*React Native\*\* as our development framework.

**Justification:**

- It supports Android and Apple devices.

- The team is familiar with JavaScript and React.

- It provides a wide range of libraries for navigation and notifications.

**Consequences:**

- The app will require third-party libraries like `react-navigation` and `react-native-push-notifications`.

- Some advanced features may require native code integration.

### **3.2 Navigation Strategy**

**Status:** Accepted

**Context:**

The app needs a simple way for users to navigate between the \*\*Home\*\*, \*\*Calendar View\*\*, and \*\*Settings\*\* screens.

**Decision:** We will implement

\*\*Bottom Tab Navigation\*\* using the `react-navigation` library with the following structure:

- Home Tab: Task List and Daily/Weekly Overview

- Calendar Tab: Calendar View

- Settings Tab: App settings

## **Justification:**

Bottom Tab Navigation provides a user-friendly and intuitive way to switch between primary app sections. It aligns with standard mobile design patterns, ensuring a familiar experience for users. Additionally, it keeps navigation accessible from all screens, reducing the number of steps needed to move between key features.

**Consequences:**

- The app will have consistent user experience across all screens.

- We need to install and configure `react-navigation` dependencies.

### **3.3 Hardware Utilization**

**Status:** Accepted

**Context:**

Our app requires hardware integration to improve user experience and productivity.

**Decision:**

We will integrate the following hardware features:

- \*\*Push Notifications\*\*: Remind users of upcoming tasks.

**Consequences:**

- The app will request user permissions for Notifications access.

- Vibration feedback will be limited to supported devices.

### **3.4 Database Storage**

**Status:** Accepted

**Context:**

The app needs to store user tasks, reminders, and completed task history.

**Decision:**

We will use React native community package **react-native-mmkv** to:

- Store tasks locally on the device.

- Encrypt sensitive user data.

**Consequences:**

- The app will work offline without needing an internet connection.

- Data will be stored securely, but syncing with external services will require additional development in the future.

## **4. Progress Report**

|  |  |
| --- | --- |
| **Highlights and Accomplishments** | * Developed the architectural framework * Determined the development framework, navigation strategy, hardware utilization and database storage. |
| **Challenges and Issues** | * Considering the team’s skill set in architecture decisions. * Managing time effectively alongside coursework. * Balancing workload among team members. |
| **Individual Contributions** | * Alessandra Nicole Claur - Contributed to design planning and technical documentation * Bryan Benedict Teruel - Designed the architectural framework and database schema * Gabriel Ira Siwa - Assisted in setting up the frontend structure * Lulubelle Fontelo - Facilitated decision-making discussions and project conceptualization. * Mitzi Vera Escartin - Compiled technical documentation, ensuring clear documentation of the system. * All - Participated in discussions, collaborated on architecture decisions, and ensured project alignment with initial planning. |
| **Team Plan for the Next Phase** | * Design wireframes and UI mock-ups for key pages * Develop a prototype using Figma |
| **Individual Plan for the Next Phase** | * **Alessandra** – Work on designing the dashboard and its user interactions. * **Bryan** – Develop wireframe and work on notification page * **Gabriel** – Focus on creating UI components and calendar page. * **Lulubelle** – Assist in integrating UI/UX mock-ups and other pages * **Mitzi** – Ensure documentation of UI/UX decisions and assist on creating UI components. |

## **5. Attribution List**

|  |  |
| --- | --- |
| **Task** | **Assigned Member** |
| Architecture Diagram | Bryan Benedict Teruel |
| Navigation Strategy | Alessandra Nicole Claur |
| Development Framework | Gabriel Ira Siwa |
| Hardware Utilization | Mitzi Vera Escartin |
| Database Storage | Lulubell Fontelo |